IMAGE REGISTRATION METHOD IMPROVEMENT

Field of the Invention

The present invention relates generally to image registration and, in particular, to determining a transformation that registers two images that are related by rotation, scale and translation.

Background

Image registration is the process of determining the correspondence between pixel elements in a pair of images that have common subject matter. In particular, image registration involves determining the parameters of a transformation that relates the pixel elements in the pair of images. Image registration is therefore an important aspect of image matching, where two images are compared for common subject matter under the assumption that some geometrical transformation exists that relates substantial portions of the two images. Image registration is also used in satellite and medical imagery where a series of partially overlapping images are obtained, and a mosaic of the individual images has to be formed to thereby form a single large image.

Image registration is also useful in camera calibration, where an image of a known object is captured and the location of that known object within the image is calculated to determine some unknown parameters of the imaging device. Yet another application of image registration is as part of a system for determining the relative video camera orientation and position between video frames in a video sequence.

A simple form of image registration may be used when the two images are related through translation only. In such a case, a matched filtering method may be used to find the translation relating the two images. Cross-correlation and phase-correlation are examples of two such matched filtering methods.

Cross-correlation may be performed in the spatial or frequency domain. Consider two images, $I_1(x,y)$, and $I_2(x,y)$ that are functions of pixel coordinates x and y. The cross-correlation of the two images $I_1(x,y)$, and $I_2(x,y)$ in the spatial domain is defined by:

$$C(x',y') = \sum_x \sum_y I_1(x,y) I_2(x+x',y+y') \tag{1}$$

If the two images $I_1(x,y)$, and $I_2(x,y)$ are related by a simple translation (Δ_x, Δ_y) whereby:

$$I_2(x,y) = I_1(x-\Delta_x, y-\Delta_y), \tag{2}$$

then the cross-correlation $C(x',y')$ has a maximum value at the translation coordinates (Δ_x, Δ_y) where:

$$C(\Delta_x,\Delta_y) = \sum_x \sum_y I_1(x,y)^2 \tag{3}$$

Thus, by calculating the cross-correlation $C(x',y')$ of the two images $I_1(x,y)$, and $I_2(x,y)$, the translation (Δ_x, Δ_y) that registers the two images $I_1(x,y)$, and $I_2(x,y)$ may be determined.

Cross-correlation is generally performed using the Fast Fourier Transform (FFT). For an image $I_n(x,y)$, the discrete Fourier transform $\Im(I)$ is defined by:

$$\Im[I_n](u,v) = \sum_{x=0}^{N_x} \sum_{y=0}^{N_y} I_n(x,y) e^{-2\pi i x u/N_x} e^{-2\pi i y v/N_y}, \tag{4}$$

where N_x and N_y are the image dimensions in the x and y dimensions respectively. The inverse discrete Fourier transform $\Im^{-1}(F)$ is defined by:

$$\mathfrak{I}^{-1}(F) = \frac{1}{N_x N_y} \sum_{u=0}^{N_x} \sum_{v=0}^{N_y} F[I](u,v) e^{2\pi i x u / N_x} e^{2\pi i y v / N_y} .$$ (5)

The FFT is a computationally efficient method of calculating the Discrete Fourier Transform $\mathfrak{I}(I)$ and its inverse $\mathfrak{I}^{-1}(F)$.

The cross-correlation C may be calculated through:

$$C = \mathfrak{I}^{-1}(\mathfrak{I}(I_1)\mathfrak{I}(I_2)^*) ,$$ (6)

where $\mathfrak{I}(I_2)^*$ denotes the complex conjugation of the Discrete Fourier Transform $\mathfrak{I}(I_2)$. Thus, taking the inverse FFT $\mathfrak{I}^{-1}()$ of the product of the FFT of one image $\mathfrak{I}(I_1)$ and the complex conjugate of the FFT of the other image $\mathfrak{I}(I_2)^*$ leads to a further image which contains the values of the cross-correlation C which is equivalent to that defined in Equation (1).

Phase-correlation C' is another matched filtering method that is often used and is defined by:

$$C' = \mathfrak{I}^{-1}(\frac{\mathfrak{I}(I_1)}{|\mathfrak{I}(I_1)|} \frac{\mathfrak{I}(I_2)^*}{|\mathfrak{I}(I_2)|})$$ (7)

That is, rather than using the discrete Fourier transforms $\mathfrak{I}(I)$ of the images $I_1(x,y)$, and $I_2(x,y)$, only the complex phase part of the discrete Fourier transforms of the images are used. If the images $I_1(x,y)$, and $I_2(x,y)$ are related by a translational offset (Δ_x, Δ_y), the phase correlation C' will have a very sharp peak at the translation coordinates (Δ_x, Δ_y) that relates the two images $I_1(x,y)$, and $I_2(x,y)$, and small values elsewhere in the phase correlation C'.

Whereas matched filtering is used when the two images $I_1(x,y)$, and $I_2(x,y)$ are related through translation only, when the two images $I_1(x,y)$, and $I_2(x,y)$ are

related by a rotation and a scale transformation, such that image $I_2(x, y)$ is a rotated and scaled version of image $I_1(x, y)$, i.e.

$$I_2(x, y) = I_1(s(x\cos\vartheta + y\sin\vartheta), s(-x\sin\vartheta + y\cos\vartheta)), \tag{8}$$

wherein s is a scale factor and ϑ is a rotation angle, the unknown rotation ϑ and scale s parameters may be determined by transforming the images $I_1(x, y)$, and $I_2(x, y)$ into a log-polar coordinate space through:

$$\rho = \frac{1}{2}\log(x^2 + y^2)$$
$$\phi = \tan^{-1}\frac{y}{x} \tag{9}$$

The translation above leads to a relationship between the images $I_1(x, y)$, and $I_2(x, y)$ in the log-polar space as follows:

$$I_2(\rho, \phi) = I_1(\rho + \log s, \phi + \vartheta), \tag{10}$$

The matched filtering methods described above may then be used to determine the scale and rotation parameters s and ϑ from the peak in the correlation C at coordinate $(\log s, \vartheta)$.

Image registration is also often applied to a pair of images $I_1(x, y)$ and $I_2(x, y)$ where the correspondence between pixel elements is not a simple transformation, such as a translation, or rotation and scale. It may be necessary to register two images $I_1(x, y)$ and $I_2(x, y)$ that are captured using different imaging devices, or by the same imaging device but where each image is captured using a different configuration of the imaging device. In such cases the transformation includes translation, rotation and scaling.

Consider two images $I_1(x, y)$, and $I_2(x, y)$ that are related by a translation as well as a rotation and a scale, such that:

$$I_2(x, y) = I_1(s(x\cos\vartheta + y\sin\vartheta) + \Delta_x, s(-x\sin\vartheta + y\cos\vartheta) + \Delta_y) \qquad (11)$$

Current methods of registering images related by translation, rotation and scale, suffer from poor signal to noise ratios over a large class of images when the transformation parameters are such that the overlap between the two images is significantly reduced. Furthermore, current methods contain a 180-degree ambiguity, which leads to computational inefficiencies. This ambiguity is the result of the fact that the Fourier magnitude of a real function is symmetric.

Summary

It is an object of the present invention to substantially overcome, or at least ameliorate, one or more disadvantages of existing arrangements.

According to a first aspect of the present disclosure, there is provided a method of determining at least rotation and scale parameters of a transformation relating two images, said method comprising the steps of:

forming a spatial domain representation of each of said images that is invariant to translation of said images;

performing Fourier-Mellin correlation between said representations;

detecting a magnitude peak in said correlation; and

determining said rotation and scale parameters from the position of said magnitude peak.

According to a second aspect of the present disclosure, there is provided a method of determining at least rotation and scale parameters of a transformation relating two images, said method comprising the steps of:

forming a multi-channel function of each of said images by applying an operator to said images, said operator being commutative within a constant to rotation and scale;

forming a representation of each of said multi-channel functions that is

5   invariant to translation of said multi-channel function;

performing Fourier-Mellin correlation between said representations;

detecting a magnitude peak in said correlation; and

determining said rotation and scale parameters from the position of said magnitude peak.

10   According to another aspect of the present disclosure, there is provided an apparatus for implementing any one of the aforementioned methods.

According to another aspect of the present disclosure there is provided a computer program product including a computer readable medium having recorded thereon a computer program for implementing any one of the methods described above.

15   Other aspects of the invention are also disclosed.

Brief Description of the Drawings

One or more embodiments of the present invention will now be described with reference to the drawings, in which:

Fig. 1 is a flow diagram of a method of determining a transformation relating

20   two images according to an embodiment of the present invention, with the transformation including rotation, scale and translation;

Fig. 2 is a schematic block diagram of a general purpose computer upon which arrangements described can be practiced;

Fig. 3 shows a more detailed flow diagram of a step of determining the rotation and scale parameters which relates the two images, and performed in the method shown in Fig. 1;

Fig. 4 shows a more detailed flow diagram of a step of determining the translation parameters which relates the two images, and performed in the method shown in Fig. 1;

Figs. 5 and 6 show more detailed flow diagrams of alternate implementations of forming a complex image from an image with real values, and are sub-steps of a step in Fig. 3;

Fig. 7 shows a more detailed flow diagram of forming a representation that is translation invariant in the spatial domain of one of the complex images, and are sub-steps of a step in Fig. 3; and

Fig. 8 shows a more detailed flow diagram of performing Fourier-Mellin correlation, which is performed in a step in Fig. 3.

Detailed Description

Where reference is made in any one or more of the accompanying drawings to steps and/or features, which have the same reference numerals, those steps and/or features have for the purposes of this description the same function(s) or operation(s), unless the contrary intention appears.

It is to be noted that the discussions contained in the "Background" section relating to current methods of registering images should not be interpreted as a representation by the present inventors or patent applicant that such methods in any way form part of the common general knowledge in the art.

Fig. 1 is a flow diagram of a method 200 of determining a transformation relating two images according to an embodiment of the present invention, with the

transformation including rotation, scale and translation. The method 200 of determining a transformation relating two images is preferably practiced using a general-purpose computer system 100, such as that shown in Fig. 2 wherein the method of Fig. 1 may be implemented as software, such as an application program, executing within the computer system 100. In particular, the steps of the method 200 of determining a transformation relating two images are effected by instructions in the software that are carried out by the computer system 100. The software may be stored in a computer readable medium, including the storage devices described below, for example. The software is loaded into the computer system 100 from the computer readable medium, and then executed by the computer system 100. A computer readable medium having such software or computer program recorded on it is a computer program product. The use of the computer program product in the computer system 100 preferably effects an advantageous apparatus for determining a transformation relating two images.

The computer system 100 is formed by a computer module 101, input devices such as a keyboard 102 and mouse 103, output devices including a printer 115 and a display device 114. A Modulator-Demodulator (Modem) transceiver device 116 is used by the computer module 101 for communicating to and from a communications network 120, for example connectable via a telephone line 121 or other functional medium. The modem 116 can be used to obtain access to the Internet, and other network systems, such as a Local Area Network (LAN) or a Wide Area Network (WAN), and may be incorporated into the computer module 101 in some implementations.

The computer module 101 typically includes at least one processor unit 105, and a memory unit 106, for example formed from semiconductor random access memory (RAM) and read only memory (ROM). The module 101 also includes an

number of input/output (I/O) interfaces including a video interface 107 that couples to the video display 114, an I/O interface 113 for the keyboard 102 and mouse 103, and an interface 108 for the modem 116 and printer 115. A storage device 109 is provided and typically includes a hard disk drive 110 and a floppy disk drive 111. A CD-ROM

5   drive 112 is typically provided as a non-volatile source of data. The components 105 to 113 of the computer module 101, typically communicate via an interconnected bus 104 and in a manner which results in a conventional mode of operation of the computer system 100 known to those in the relevant art.

Typically, the application program is resident on the hard disk drive 110 and

10   read and controlled in its execution by the processor 105. Intermediate storage of the program and any data fetched from the network 120 may be accomplished using the semiconductor memory 106, possibly in concert with the hard disk drive 110. In some instances, the application program may be supplied to the user encoded on a CD-ROM or floppy disk and read via the corresponding drive 112 or 111, or alternatively may be

15   read by the user from the network 120 via the modem device 116. The term "computer readable medium" as used herein refers to any storage or transmission medium that participates in providing instructions and/or data to the computer system 100 for execution and/or processing.

The present invention also implicitly discloses a computer program, in that it

20   would be apparent to the person skilled in the art that the individual steps of the preferred method 100 described herein are able to be put into effect by computer code.

A method is disclosed that may be used for determining a transformation relating two images $I_1(x,y)$ and $I_2(x,y)$ that are assumed to be related by rotation, scale and translation, such that:

25

$$I_2(x,y) = I_1(s(x\cos\vartheta + y\sin\vartheta) + \Delta_x, s(-x\sin\vartheta + y\cos\vartheta) + \Delta_y) \quad (12)$$

where s is a scale factor, ϑ is a rotation angle, and (Δ_x, Δ_y) is a translation. The method comprises two stages – a first which determines the unknown scale and rotation transformation up to a one-hundred and eighty degree ambiguity in the rotation angle ϑ, and a second which determines the translation (Δ_x, Δ_y) and resolves the angular ambiguity. The Fourier transform of the scaled, rotated and shifted image $I_2(x, y)$ is related to the other image $I_1(x, y)$ through

$$\Im[I_2](u,v) = \frac{1}{|s|^2} \Im[I_1](\frac{u\cos\vartheta + v\sin\vartheta}{s}, \frac{-u\sin\vartheta + v\cos\vartheta}{s})e^{2\pi i u \Delta_x}e^{2\pi i v \Delta_y}. \quad (13)$$

In the first stage, by taking the magnitude of the Fourier transform $\Im[I_2]$, a translation invariant of the image is formed,

$$\left|\Im[I_2](u,v)\right| = \frac{1}{|s|^2}\left|\Im[I_1](\frac{u\cos\vartheta + v\sin\vartheta}{s}, \frac{-u\sin\vartheta + v\cos\vartheta}{s})\right|. \quad (14)$$

The translation invariant is not dependent on the translation (Δ_x, Δ_y) of the image. Performing a log-polar transformation of the Fourier magnitude leads to a simple linear relationship between the Fourier magnitudes of the two images as follows:

$$\left|\Im[I_2](R,\Phi)\right| = \frac{1}{|s|^2}\left|\Im[I_1](R - \log s, \Phi + \vartheta)\right|. \quad (15)$$

A correlation between a log-polar resampling of the Fourier magnitude of the two images contains a peak at $\log s$ and ϑ, thereby allowing the unknown scale s and rotation angle ϑ parameters relating the two images $I_1(x, y)$ and $I_2(x, y)$ to be determined, with the rotation angle ϑ having a 180-degree ambiguity. This ambiguity is the result of the fact that the Fourier magnitude of a real function is symmetric.

The second stage of the method of determining the transformation relating the two images $I_1(x,y)$ and $I_2(x,y)$ starts by undoing the now known scale and rotation translations for both possible rotation angles ϑ for the second image $I_2(x,y)$ to produced a partially registered image. The partially registered image is then correlated with the first image $I_1(x,y)$ to determine the unknown translation (Δ_x, Δ_y) between the two images $I_1(x,y)$ and $I_2(x,y)$. The rotation angle ϑ that gives the best spatial correlation between the partially registered image and the first image $I_1(x,y)$ is considered to be the correct rotation angle ϑ. The complete transformation relating the two images $I_1(x,y)$ and $I_2(x,y)$ is now known.

Referring again to Fig. 1 where the flow diagram of method 200 of determining a transformation relating two images $I_1(x,y)$ and $I_2(x,y)$ according to the present invention is shown. The method 200 receives in step 205 as input the two images $I_1(x,y)$ and $I_2(x,y)$. The images $I_1(x,y)$ and $I_2(x,y)$ are assumed to have a substantial overlap in image content. The images $I_1(x,y)$ and $I_2(x,y)$ are functions with real values, which are typically represented by an array of values between 0 and a predetermined maximum value, commonly 1 or 255. The images $I_1(x,y)$ and $I_2(x,y)$ are typically retrieved from the storage device 109 (Fig. 2) in step 205. However, the images $I_1(x,y)$ and $I_2(x,y)$ may also be received from the network 120 or from an imaging device (not illustrated) connected to the computer system 100.

Image registration is next performed in steps 210 and 270. In particular, the rotation and scale parameters, ϑ and s respectively, which relate the two images $I_1(x,y)$ and $I_2(x,y)$ are determined in step 210, and the translation (Δ_x, Δ_y) is determined in step 270. Method 200 ends in step 290 where an output is produced to the display device 114 containing two substantially aligned images $I_1''(x,y)$ and $I_2(x,y)$.

Alternatively the scale factor s, rotation angle ϑ, and translation (Δ_x, Δ_y) are output on the display device 114.

Fig. 3 shows a more detailed flow diagram of step 210 (Fig. 1) where the rotation and scale parameters, ϑ and s respectively, which relate the two images $I_1(x,y)$ and $I_2(x,y)$ are determined. Step 210 of the preferred implementation starts in sub-step 212 where the processor 105 (Fig. 2) forms a multi channel function from the images $I_1(x,y)$ and $I_2(x,y)$. In the preferred implementation the processor 105 forms complex images $\bar{I}_1(x,y)$ and $\bar{I}_2(x,y)$ from the images $I_1(x,y)$, and $I_2(x,y)$, such that when each complex image $\bar{I}_n(x,y)$ is Fourier transformed, a non-Hermitian result with a non-symmetric Fourier magnitude is produced. Therefore, using a complex image $\bar{I}_n(x,y)$ as the input to the Fourier-Mellin correlation that follows, a 180-degree ambiguity that would otherwise exist is removed.

The complex images $\bar{I}_1(x,y)$ and $\bar{I}_2(x,y)$ are formed by applying an operator $\gamma\{\ \}$ to the images $I_1(x,y)$, and $I_2(x,y)$, where the operator $\gamma\{\ \}$ is commutative within a constant to rotation and scale, ie.

$$T_{\beta,s}[\gamma\{f(x,y)\}] = g(\beta,s)\gamma\{T_{\beta,s}[f(x,y)]\};\tag{16}$$

where β and s are rotation and scale factors, $T_{\beta,s}$ is a rotation-scale transformation, and g is some function of rotation β and scale s.

Examples of the operator $\gamma\{\ \}$ include:

$$\gamma\{f(x,y)\} = f(x,y) + if^2(x,y);\tag{17}$$

$$\gamma\{f(x,y)\} = \frac{\partial f}{\partial x} + i\frac{\partial f}{\partial y};\ \text{and}\tag{18}$$

$$\gamma\{f(x,y)\} = \left(\frac{\partial f}{\partial x} + i\frac{\partial f}{\partial y}\right)^2 . \tag{19}$$

Preferred steps of forming a complex image $\bar{I}_n(x,y)$ from an image $I_n(x,y)$ are described below with reference to Figs. 5 and 6.

The multi channel functions formed in sub-step 212, those being the complex images $\bar{I}_1(x,y)$ and $\bar{I}_2(x,y)$ in the preferred implementation, are then each processed by the processor 105 in sub-step 214 to form a representation $T_1(x,y)$ and $T_2(x,y)$ of each of the two complex images $\bar{I}_1(x,y)$ and $\bar{I}_2(x,y)$, where representations $T_1(x,y)$ and $T_2(x,y)$ are translation invariant in the spatial domain. It is understood that translations of images cause cropping of the images. Translation invariant therefore means substantially translation invariant, as the cropping introduces changes to the images separate to changes introduced by the translation.

Sub-step 214 is followed by sub-step 216 where the processor 105 performs Fourier-Mellin correlation on the representations $T_1(x,y)$ and $T_2(x,y)$ of the two complex images $\bar{I}_1(x,y)$ and $\bar{I}_2(x,y)$, thereby producing a phase correlation image in which possible rotation and scaling that relate the input images $I_1(x,y)$ and $I_2(x,y)$ are represented by isolated peaks. Fourier-Mellin correlation is described in more detail below with reference to Fig. 8.

Because the representations $T_1(x,y)$ and $T_2(x,y)$ are translation invariant in the spatial domain, the Fourier-Mellin correlation produces superior results for images $I_1(x,y)$ and $I_2(x,y)$ that are related by a wide range of translation, rotation and scale factors. Such superior results typically include increased matched filter signal to noise ratio for images that are related by rotation, scale and translation transformations, and

enhanced discrimination between images that are not related by a rotation, scale and translation transformations.

Method 200 continues to sub-step 218 where the processor 105 detects the location of a magnitude peak within the phase correlation image. The location of the magnitude peak is interpolated through quadratic fitting, thereby detecting the location of the magnitude peak to sub-pixel accuracy.

The processor 105 then determines in sub-step 220 whether the detected magnitude peak has a signal to noise ratio that is greater than a predetermined threshold. The threshold used in the preferred implementation is 1.5.

If it is determined that the magnitude peak has a signal to noise ratio that is greater than the predetermined threshold, the processor 105 uses in sub-step 222 the location of the magnitude peak to determine the scale s and rotation angle ϑ parameters which relate the two images $I_1(x, y)$ and $I_2(x, y)$. If the magnitude peak is at location (ζ, α), then the scale s and rotation angle ϑ parameters which relate the two images $I_1(x, y)$ and $I_2(x, y)$ are:

$$s = e^{-a\zeta} \text{ ,and} \tag{20}$$

$$\vartheta = -\frac{2\pi\alpha}{Q}. \tag{21}$$

where a and Q are constants related to the log-polar resampling step 630 of the Fourier-Mellin correlation described below with reference to Fig. 8.

If it is determined in sub-step 220 that the peak has a signal to noise ratio that is not greater than the predetermined threshold, then it is concluded that the images $I_1(x, y)$ and $I_2(x, y)$ are not related, and the method 200 ends in sub-step 250.

Fig. 4 shows a more detailed flow diagram of step 270 (Fig. 1) where the translation (Δ_x, Δ_y), which relates the two images $I_1(x,y)$ and $I_2(x,y)$, is determined. In sub-step 280 the rotation and scale transformations, determined in sub-step 222 (Fig. 3), are performed on image $I_1(x,y)$, thereby undoing those transformations to form image $I_1'(x,y)$. Optionally the transformations may be performed on the complex image $\bar{I}_1(x,y)$. The rotated and scaled transformed image $I_1'(x,y)$ and the original image $I_2(x,y)$ are then correlated in sub-step 282, using phase correlation, to produce another correlation image. The position of a magnitude peak in this correlation image will generally correspond to the translation (Δ_x, Δ_y) relating images $I_1(x,y)$, and $I_2(x,y)$. Accordingly, in sub-step 284 the processor 105 detects the location of the magnitude peak within the correlation image.

The processor 105 then, in sub-step 286, uses the location of the magnitude peak to determine the translation parameters (Δ_x, Δ_y) which relate the two images $I_1'(x,y)$ and $I_2(x,y)$. The same translation parameters (Δ_x, Δ_y) also relate the original two images $I_1(x,y)$ and $I_2(x,y)$. If the magnitude peak is at location (x_0, y_0), then the translation (Δ_x, Δ_y) is $(-x_0, -y_0)$. Thus, the unknown scale s and rotation angle ϑ parameters have been determined in sub-step 222, and the unknown translation parameters (Δ_x, Δ_y) have been determined in sub-step 286. Those parameters are passed to step 290 (Fig. 1) for producing the output.

Fig. 5 shows a more detailed flow diagram of a first implementation of sub-step 212 (Fig. 3) where the complex image $\bar{I}_n(x,y)$ is formed from the image $I_n(x,y)$. In sub-step 320 the image $I_n(x,y)$ is convolved with a complex kernel function k by the processor 105. The convolution may be performed in the spatial domain or through the

standard technique of multiplication in the Fourier domain. The complex kernel function k used in sub-step 320 is that with a Fourier transform $K = \Im(k)$ of:

$$K(u,v) = \frac{u+iv}{|u+iv|}.$$
(22)

An alternative complex kernel function k' that may be used in sub-step 320 is one with a Fourier transform $K' = \Im(k')$ of:

$$K'(u,v) = u + iv.$$
(23)

The result of the convolution $((I*k)$, where $*$ denotes convolution,) is normalised in sub-step 330 to have unit magnitude,

$$\Gamma = \frac{I*k}{|I*k|},$$
(24)

Finally, the normalised result of the convolution Γ is multiplied with the original image $I_n(x,y)$ in sub-step 340 to form the complex image $\bar{I}_n(x,y)$. The complex image $\bar{I}_n(x,y)$ has the same magnitude as the original image $I_n(x,y)$, but each point in the complex image $\bar{I}_n(x,y)$ has an associated phase generated by the convolution in sub-step 320. For the kernels k and k' given in Equations (22) and (23), the associated phase encodes a quantity related to the gradient direction of the image $I_n(x,y)$.

Fig. 6 shows a more detailed flow diagram of a second (alternate) implementation of sub-step 212 (Fig. 3) where the complex image $\bar{I}_n(x,y)$ is formed from the image $I_n(x,y)$. In sub-step 420 the processor 105 applies a non-linear operator to the image $I_n(x,y)$ to produce a complex image. The non-linear operator applied in sub-step 420 is the energy operator, which may be described by

$$E[I] = ID^2I - (DI)^2,$$ (25)

where D is the derivative operator

$$D = \frac{\partial}{\partial x} + i\frac{\partial}{\partial y}.$$ (26)

An alternative non-linear operator that may be applied in sub-step 420 to produce the complex image is the uni-modular energy operator:

$$E'[I] = ID'^2I - (D'I)^2,$$ (27)

where D' is the uni-modular derivative operator. The uni-modular derivative operator D' may be described as an operation in the Fourier domain as follows:

$$D'(I) = \mathfrak{I}^{-1}\left[\mathfrak{I}[I]\frac{u + iv}{|u + iv|}\right].$$ (28)

Preferably, in sub-step 430 which follows sub-step 420, the result of the non-linear operator applied to image $I_n(x,y)$ is normalised to unit modulus, and the result of this normalisation is multiplied by the original image $I_n(x,y)$ in sub-step 440 to form the complex image $\bar{I}_n(x,y)$. Alternatively, the result of the non-linear operator applied to image $I_n(x,y)$, hence the output of sub-step 420 may be used as the complex image $\bar{I}_n(x,y)$.

Fig. 7 shows a more detailed flow diagram of forming a representation $T_n(x,y)$ of one of the complex images $\bar{I}_n(x,y)$ that is translation invariant in the spatial domain performed in sub-step 214 (Fig. 3). Sub-step 214 receives as input the complex images $\bar{I}_n(x,y)$ formed in sub-step 212. Each complex image $\bar{I}_n(x,y)$ is first Fourier transformed by the processor 105 in sub-step 520, using the FFT, thereby producing an

image consisting of complex values. This image is separated in sub-step 530 into two separate images, those being a magnitude image containing the magnitudes of the complex values of the Fourier transform, and a phase image containing the phases of the complex values of the Fourier transform. In sub-step 560 a function is applied to the magnitude image, with the function being commutative within a constant to rotation and scale. In the preferred implementation the magnitude image is multiplied by a ramp function to perform high-pass filtering of the magnitude image. In sub-step 570 an operator is applied to the phase image to take the second or higher derivative of the phase, which is a translation invariant. In the preferred implementation the Laplacian operator is used.

Sub-step 214 continues to sub-step 580 where the modified magnitude image produced from sub-step 560, and the result of taking the Laplacian of the phase image produced from sub-step 570 are combined through

$$|F| + iA\nabla^2\varphi \tag{29}$$

where $|F|$ is the modified magnitudes of the Fourier transform of the complex images $\bar{I}_n(x,y)$, $\nabla^2\varphi$ is the Laplacian of the phase image of the Fourier transform , and A is a scaling constant set to:

$$A = \max(|F|)/\pi. \tag{30}$$

The scaling constant A ensures that the recombined Fourier magnitude and phase information are roughly of equal magnitude.

The result of the combining of the modified magnitude image and the result of taking the Laplacian of the phase image is then inverse Fourier transformed in sub-step 590, thereby producing the representation $T_n(x,y)$ that is translation invariant in the spatial domain.

Other translation invariants of the Fourier magnitude and phase may be used in place of sub-steps 560 and 580, such as:

the modulus squared of the Fourier magnitude;

the logarithm of the Fourier magnitude;

the Laplacian of the logarithm of the Fourier transform; or

operators such as

$$\left(\left(\frac{\partial^2}{\partial u^2}+\frac{\partial^2}{\partial v^2}\right)+i\left(\frac{\partial^2}{\partial u\partial v}-\frac{\partial^2}{\partial v\partial u}\right)\right)(\log F) \tag{31}$$

and

$$\left(\left(\frac{\partial^2}{\partial u^2}-\frac{\partial^2}{\partial v^2}\right)+i\left(\frac{\partial^2}{\partial u\partial v}+\frac{\partial^2}{\partial v\partial u}\right)\right)(\log F) \tag{32}$$

where the logarithm of a complex number is defined as

$$\log F = \log|F| + i\varphi . \tag{33}$$

Preferably, the translation invariant is free of 180° ambiguity with respect to rotation. This holds for all translation invariants listed above, except the modulus squared of the Fourier magnitude.

Fig. 8 shows a more detailed flow diagram of the sub-step of performing in sub-step 216 the Fourier-Mellin correlation on the representations $T_1(x,y)$ and $T_2(x,y)$ that are translation invariant in the spatial domain. In sub-step 630 each of the representations $T_1(x,y)$ and $T_2(x,y)$ is resampled to the log-polar domain. In order to resample to the log-polar domain, it is necessary to specify a resolution within the log-polar domain. If the original images $I_1(x,y)$ and $I_2(x,y)$ are N pixels wide by M pixels high, hence the coordinate x varies between 0 and N-1, while the y-coordinate

varies between 0 and M-1, then the centres of the representations $T_1(x,y)$ and $T_2(x,y)$ which are translation invariant in the spatial domain are located at $(c_x, c_y) = (\text{floor}(N/2), \text{floor}(M/2))$. Log-polar resampling to an image having dimensions P pixels by Q pixels in log-polar space is performed relative to this centre. To avoid a singularity at the origin, it is necessary to ignore a disc of radius r_{min} pixels around the centres of the representations $T_1(x,y)$ and $T_2(x,y)$. While ignoring this disc, a point (x,y) in the log-polar plane is determined by interpolating the translation invariant image at the point (x,y) as follows:

$$x = c_x + \cos\frac{2\pi j}{Q} r_{min} e^{ai}$$
$$y = c_y + \sin\frac{2\pi j}{Q} r_{min} e^{ai}$$

(34)

wherein

$$a = \frac{\log r_{max}/r_{min}}{P-1}$$

(35)

and

$$r_{max} = \max\{M/2, N/2\}$$

(36)

denotes the maximum radius in the spatial domain that the log-polar image extends to. Common values of the constants r_{min}, P and Q are:

$$P = Q = (M+N)/2, \text{ and}$$

(37)

$$r_{min} = 5.$$

(38)

In sub-step 640 the processor 105 performs the Fourier transform on each of the resampled representations $T_1(x,y)$ and $T_2(x,y)$. Sub-step 640 is followed by sub-step 650 where the processor 105 performs a complex conjugation on the second resampled

representation $T_2(x, y)$. Both Fourier transforms are then normalised in sub-step 660 so that each has unit magnitude by dividing each complex element of each Fourier transform by the magnitude of the complex element. The normalised Fourier transforms are then multiplied in sub-step 670, and the result of the multiplication is then inverse

5   Fourier transformed in sub-step 680. A phase correlation image results.

The above describes the preferred implementations of the present disclosure and it will be understood that there are a number of modifications and/or changes that can be made thereto without departing from the scope and spirit of the disclosure. In particular, sub-step 212 (Fig. 3) where multi-channel functions are formed may be

10   neglected while still retaining sub-step 214 where representations that are invariant to translations in the spatial domain are formed.

Method 200 has been described in terms of operations on images $I_1(x, y)$ and $I_2(x, y)$ that have only one component. Multi-component images, such as colour images and images from multi-spectral devices can also be registered with respect to

15   each other using method 200. This is achieved by forming a single component image from each of the multi-component images through some algebraic combination of the components, and registering the two single component images $I_1(x, y)$ and $I_2(x, y)$ thus generated. Alternatively it may be achieved by registering each component of the first multi-component image against each component of the other multi-component

20   image separately.

In the context of this specification, the word "comprising" means "including principally but not necessarily solely" or "having" or "including", and not "consisting only of". Variations of the word "comprising", such as "comprise" and "comprises" have correspondingly varied meanings.